



## C1 - 7 条件判断 - if 语句

### // 导语

上次课我们学会了如何使用 C++ 语言描述常见的问题，例如如何说明一个数字是 7 的倍数，如何说明一个变量里存放的是一个字母，又如何说明一个年份是闰年。这些问题的描述过程非常重要，今天我们就可以拿它来进行条件判断。

---

### // 学习目标

- 了解基本程序结构
  - 学会使用单分支 if 语句进行简单判断
  - 掌握多分支 if 语句的嵌套使用
- 

### // 程序结构

C++ 的程序结构分为 3 种：

- 顺序结构：指令按照程序出现的顺序执行
- 分支结构：程序步骤出现分支，根据条件选择其中一支
- 循环结构：重复执行一句或多句指令

今天我们学习的是用来进行条件判断的“分支结构”。  
如果条件满足，那么... ..

说到“如果”，你知道用什么单词来表示它吗？



## // 单分支结构 - if 语句

有些问题只需要判断一个条件是否满足即可，一般称之为单分支结构，如判断一个数字  $x$  是不是奇数，可以怎么做呢？

首先我们知道奇数是不能被 2 整除的，其表达式为  $x \% 2 \neq 0$ ，接下来我们把这句话放在条件判断语句 —— if 语句里，代码可以这样写：

```
#include <iostream>
using namespace std ;
int main(){
    int x ;
    cin >> x ;
    if ( x % 2 != 0 ) // 只判断是不是奇数，无需判断否则的情况
        cout << "是奇数" ;
    return 0 ;
}
```

如果输入的数字  $x$  是奇数就会输出“是奇数”字样，否则什么也不做。

---

## // 双分支结构 - if ... else ...语句

上述问题中，如果添加一个“否则”的分支，代码还可以这样写：

```
if ( x % 2 != 0 )
    cout << "是奇数" ;
else cout << "是偶数" ; // 添加了偶数情况对应的操作
```

可以准确地说如果是奇数就输出“是奇数”，否则一定会输出“是偶数”。

注意：if ... else ... 语句会在满足条件时默认执行后面的第一条指令，如果需要满足一个条件执行多条指令，那么需要加上大括号 “{ }”，例如：

```
if ( x % 2 != 0 )
{
    y = x ;
    cout << "是奇数" ;
}
else
{
    z = x ;
    cout << "是偶数" ;
}
```

## // 多分支结构 - if ... else ...嵌套语句

若一个问题中出现多种情况时，此时可以使用多分支，例如判断整数  $x$  是正负数，判断正负数时只有三种情况：

情况 1:  $x > 0$  , 正数;

情况 2:  $x = 0$  , 零; (0 既不是正数也不是负数)

情况 3:  $x < 0$  , 负数;

因此可以写出对应的代码：

```
if ( x > 0 ) cout << "正数";  
if ( x == 0 ) cout << "零";  
if ( x < 0 ) cout << "负数";
```

当然，还可以换一种思路写：

```
if ( x > 0 ) cout << "正数";  
else {  
    if ( x == 0 ) cout << "零";  
    else cout << "负数";  
}
```

**注意：**在代码中发现多个 if 和 else 的时候可以通过观察缩进来判断他们的配对情况，如果代码中没有缩进格式，那么可以通过找 else 最近的没有配对的 if 进行配对阅读。

所以当我们自己在写代码的时候一定要让配对的 if...else... 缩进统一哦！这样也方便自己和老师阅读。

这两种方案看起来结果都一样，但是严格来讲第二种方案更严谨些，因为第一种方案如果不是判断互斥的条件，很有可能多种条件同时满足，而判断正负数时恰巧是绝对互斥的情况（绝对互斥是指满足一个条件同时不能满足其他的所有条件），因此在这里也可以使用，对于**第二种方案**，其**代码可以简化为**：

```
if ( x > 0 ) cout << "正数";  
else if ( x == 0 ) cout << "零";  
else cout << "负数";
```